CLAIMS

WHAT IS CLAIMED IS:

1. A method of providing information from a database comprising the acts of:

5   determining that a relational expression includes a join operator having:

a first operand which includes a first instance of a relation based on information stored in said database, said relation having a set of rows and a set of columns;

10   a second operand including a second instance of said relation; and

a join predicate which is, or conjunctively includes, an equality comparison between one or more columns of said first instance of said relation and corresponding one or more columns of said second instance of said relation;

15   .   segmenting said relation based on said one or more columns to produce one or more segments of said relation; and

applying said join operator separately to each of said segments.

2. The method of claim 1, further comprising the act of identifying said

20   relational expression as a fragment of a larger expression.

3. The method of claim 1, further comprising the acts of:

creating a first expression tree representative of said relational expression; and

25   creating a second expression tree based on said first expression. tree, said second expression tree including a node having:

data indicative of said relation;

data indicative of said one or more columns; and

data indicative of said join operator.

4. The method of claim 3, further comprising the act of:

determining that second expression can be evaluated more efficiently than said first expression tree.

5. The method of claim 1, wherein said join operator comprises one of: inner join, semijoin, or anti-semijoin.

6. The method of claim 1, further comprising the act of:

determining that said first operand further includes a filter which modifies said first instance of said relation according to a second predicate.

7. The method of claim 1, further comprising the act of:

determining that said first operand further includes an aggregate operation which specifies the computation of a value based on one or more rows of said relation.

8. The method of claim 1, further comprising the act of successively spooling each of said segments, wherein said applying act comprises applying said join operator to the segment in said spool.

9. The method of claim 1, wherein said relation comprises a stored table.

10. The method of claim 1, wherein said relation comprises a sub-expression based on one or more stored tables, and wherein said method further comprises the act of evaluating said sub-expression to produce said relation.

11. The method of claim 1, further comprising the act of compiling a query to produce said relational expression.

12. The method of claim 1, wherein said segmenting act comprises:

5 sorting said relation;

spooling a first row of the sorted relation; and

spooling successive rows of the sorted relation until a row is encountered having values in said one or more columns that differ from the values in said one or more columns of said first row.

10

13. A computer-readable medium having computer-executable instructions to perform the method of claim 1.

14. A method of providing information from a database comprising the

15 . acts of:

determining that a relational expression includes:

a first join operator having:

a first operand which includes a first instance of a first relation based on information stored in said database, said first relation having a

20 set of rows and a set of columns;

a second operand including a second instance of said first relation; and

a first predicate which is, or conjunctively includes, an equality comparison between one or more columns of said first instance of

25 said first relation and corresponding one or more columns of said second instance of said first relation; and

a second join operator having:

a third operand which includes the result of said first join operator;

a fourth operand including a second relation based on information stored in said database; and

5   a second predicate;

performing a join of said second relation and said first relation using said second predicate to produce an intermediate result;

segmenting said intermediate result based on said one or more columns to produce one or more segments of said intermediate result; and

10   applying said first join operator including said first predicate separately to each of said segments.

15. The method of claim 14, wherein said segmenting act is further based on a key of said second relation.

15 .

16. The method of claim 14, wherein said segmenting act is further based on all columns of said second relation.

17. The method of claim 14, further comprising the act of:

20   determining that said second predicate references only columns in the set of columns consisting of:

said one or more columns; and

columns in said second relation.

25   .   18. The method of claim 14, further comprising the acts of:

removing conjuncts from said second predicate that reference columns other than:

said one or more columns; and

columns in said second relation; and

removing rows from the result of said applying step that do not satisfy the removed conjuncts.

5   19. The method of claim 14, further comprising the act of:

comparing the predicted efficiency of at least two procedures of evaluating said relational expression.

20. The method of claim 14, further comprising the act of identifying

10   said relational expression as a fragment of a larger expression.

21. The method of claim 14, wherein either said first join operator or said second join operator comprises one of: inner join, semijoin, or anti-semijoin.

15 .   22. The method of claim 14, further comprising the act of:

determining that said first operand further includes a filter which modifies said first instance of said first relation according to a third predicate.

23. The method of claim 14, further comprising the act of:

20   determining that said first operand further includes an aggregate operation which specifies the computation of a value based on one or more rows of said first relation.

24. The method of claim 14, further comprising the act of successively

25   spooling each of said segments, wherein said applying act comprises applying said first join operator to the segment in said spool.

25. The method of claim 14, wherein said first relation comprises a stored table.

26. The method of claim 14, wherein said first relation comprises a sub-expression based on one or more stored tables, and wherein said method further comprises the act of evaluating said sub-expression to produce said first relation.

27. The method of claim 14, wherein said segmenting act comprises:

        sorting said intermediate result;

        spooling a first row of the sorted intermediate result; and

        spooling successive rows of the sorted intermediate until a row is encountered having values in said one or more columns that differ from the values in said one or more columns of said first row.

28. A computer-readable medium having computer-executable instructions to perform the method of claim 14.

29. A method of providing information from a database comprising the acts of:

        determining that a first relational expression includes a join operator having:

        a first operand which includes a first relation based on information stored in said database;

        a second operand which includes an operator calling for a relational fragment to be applied successively to segments of said second relation, said second relation being based on information stored in said database, said segments of said second relation being based on distinct values of one or more columns of said second relation; and

a predicate;

creating a second relational expression that includes:

an operator calling for said relational fragment to be applied successively to segments of a third relation, said segments of said third relation being based on distinct values of said one or more columns and all columns of said first relation, said third relation being the result of a join operator having:

a third operand which includes said first relation;

a fourth operand which includes said second relation; and

said predicate; and

evaluating said second relational expression.

30. The method of claim 29, further comprising the act of:

determining that said predicate references only columns in the set of columns consisting of:

said one or more columns; and

columns in said first relation.

31. The method of claim 29, further comprising the acts of:

removing conjuncts from said predicate that reference columns other than:

said one or more columns; and

columns in said first relation; and

removing rows from the result of said evaluting step that do not satisfy the removed conjuncts.

32. The method of claim 29, wherein said join operator comprises one of: inner join, semijoin, or anti-semijoin.

33. A computer-readable medium having computer-executable instructions to perform the method of claim 29.

5   34. A system for producing results from a database comprising:

a spooling module which receives a relation comprising a set of rows and a set of columns, and which copies into a memory location one or more rows of said relation having common values for one or more columns of said relation; and

an application module which invokes a relational procedure, 10   wherein said relational procedure performs a relational operation on rows in said memory location to produce one or more result rows, said application module invoking said spooling module when said relational procedure is unable to produce result rows.

35. The system of claim 34, further comprising:

15   a query compiler which receives a query and produces an expression tree based on said query, said relational procedure being based on said expression tree.

36. The system of claim 35, wherein said expression tree comprises a 20   join operator having:

a first operand which includes a first instance of a relation, said relation having a set of rows and a set of columns;

a second operand including a second instance of said relation; and

a first predicate;

25   said system further comprising:

an optimization module which receives said expression tree and determines that said predicate is, or conjunctively includes, an equality comparison between one or more columns of said first instance of said relation and corresponding

one or more columns of said second instance of said relation, said one or more columns of said first and second instances of said relation being the same columns as said one or more columns of said relation.

37. The system of claim 34, wherein said relational procedure comprises a join.

38. The system of claim 37, wherein said join comprises one of: inner join, semijoin, or anti-semijoin.

39. A method of evaluating a semijoin having a first operand, a second operand, and a predicate, said method comprising the acts of:

determining that said first and second operands each comprise first and second instances, respectively, of a common relation;

determining that said predicate is, or conjunctively includes, an equality comparison between one or more columns of said first instance of said relation and corresponding columns of said second instance of said relation;

segmenting said common relation based on said one or more columns to produce one or more segments of said common relation; and

performing said semijoin separately on each of said segments.

40. The method of claim 39, further comprising the act of spooling each of said segments in a memory location, wherein said performing act comprises applying said semijoin successively to the segments spooled in said memory location.

41. The method of claim 39, further comprising the act of compiling a SQL query which includes an EXISTS clause to produce a relational expression that includes said semijoin.

42.   A   computer-readable   medium   having   computer-executable instructions to perform the method of claim 39.

5   43. A method of evaluating an anti-semijoin having a first operand, a second operand, and a predicate, said method comprising the acts of:

determining that said first and second operands each comprise first and second instances, respectively, of a common relation;

determining that said predicate is, or conjunctively includes, an
10   equality comparison between one or more columns of said first instance of said relation and corresponding columns of said second instance of said relation;

segmenting said common relation based on said one or more columns to produce one or more segments of said common relation; and

performing said anti-semijoin separately on each of said
15   segments.

44. The method of claim 43, further comprising the act of spooling each of said segments in a memory location, wherein said performing act comprises applying said anti-semijoin successively to the segments spooled in said memory location.

20

45. The method of claim 43, further comprising the act of compiling a SQL query which includes a NOT EXISTS clause to produce a relational expression that includes said anti-semijoin.

25   46.   A   computer-readable   medium   having   computer-executable instructions to perform the method of claim 43.

47. A method of sorting rows of a database table according to a first set of one or more columns, said rows having been sorted on a second set of one or more columns, said method comprising the acts of:

segmenting said database table based on said second set of columns to produce segments of said database table, each of said segments comprising rows having common values in said second set of columns; and

separately sorting each of said segments based on the values in said first set of columns.

48. The method of claim 47, wherein said segmenting act is performed without spooling rows of said segments.

49. The method of claim 47, wherein said segmenting act comprises:

identifying a first row of said database table, said first rows having first values in said second set of columns; and

identifying a second row of said database table, said second row having said first values in said second set of columns, said database table having an order, said second row being the last occurring row in said order having said first values in said second set of columns.

50. The method of claim 47, wherein said act of separately sorting comprises:

invoking a first function which sorts rows in first one of said segments;

invoking a second function which identifies a next one of said segments; and

repeating said acts of invoking said first and second function until all of said segments have been exhausted.

51. The method of claim 47, further comprising the act of:

creating an expression tree including a GbApply operator having:

a first child sub-tree specifying said database table as input

5   to said GbApply operator;

segmentation data indicating that said database table is to

be segmented on said second set of columns; and

a second child sub-tree specifying a relational fragment to

be performed on each of said segments, said relational fragment specifying a sort of the

10   rows of said segments on said first set of columns.

52.   A   computer-readable   medium   having   computer-executable

instructions to perform the method of claim 47.

15 .   53. In a database system which performs operations on a table having

rows and columns, a method of identifying a row having a superlative value in a first of

said columns from among a set of rows having a common value in a second of said

columns, said method comprising the acts of:

segmenting said rows based on the values in said second column

20   to produce groups of one or more rows, wherein all of the rows in a first of said groups

have a common value in said second column;

sorting the rows in said first group based on the values in said

first column; and

identifying the first or last row in said first group.

25

54. The method of claim 53, wherein said superlative value comprises a

maximum value, and wherein said identifying act comprises identifying the last row in

said first group.

55. The method of claim 53, wherein said superlative value comprises a minimum value, and wherein said identifying act comprises identifying the first row in said first group.

56. The method of claim 53, wherein said segmenting act comprises:

sorting said rows based on the values in said second column; and

following said sorting act, identifying the first and last rows which have said common value in said second column.

57. The method of claim 53, further comprising the act of:

creating an expression tree including a GbApply operator having:

a first child sub-tree specifying said table as input to said GbApply operator;

segmentation data indicating that said database table is to be segmented on said second set of columns; and

a second child sub-tree specifying a relational fragment to be performed on each of said segments, said relational fragment specifying the selection of either the first or last of the rows in a segment.

58. A computer-readable medium having computer-executable instructions to perform the method of claim 53.

59. A system for evaluating a database query comprising:

a query compiler which receives the query and produces an expression tree based on the query;

an analysis module which identifies an expression tree having a join operator which includes:

a first operand which includes a first instance of a relation based on information stored in said database, said relation having a set of rows and a set of columns;

a second operand including a second instance of said

5   relation; and

a first predicate which is, or conjunctively includes, an equality comparison between one or more columns of said first instance of said relation and corresponding one or more columns of said second instance of said relation; and

an evaluation module which segments said relation according to

10   the values in said one or more columns and which applies said join operator separately to each segment of said relation.

60. The system of claim 59, further comprising:

a spooling module which spools rows of each segment in a

15   designated memory location;
wherein said evaluation module applies said join operator to the rows stored in said designated memory location.

61. The system of claim 59, wherein said evaluation module does not

20   spool the segments of said relation.

62. The system of claim 59, wherein said join operator comprises one of: inner join, semijoin, or anti-semijoin.

25   63. The system of claim 59, wherein said analysis module determines that said first operand further includes a filter which modifies said first instance of said relation according to a second predicate.

64. The system of claim 59, wherein said analysis module determines that said first operand further includes an aggregate operation which specifies the computation of a value based on one or more rows of said relation.

65. The system of claim 59, wherein said relation comprises a stored table.

66. The system of claim 59, wherein said relation comprises a sub-expression based on one or more stored tables, and wherein said system further includes a module which evaluates said sub-expression to produce said relation.